A7 communicating the hypertext document to a browser for displaying said inlaid comments.

REMARKS

Claims 1-8 and 9-11 and 13-40 remain in the case, claims 9 and 12 having been cancelled without prejudice.

Claims 1, 9-10, 12, 16-20, 33, 40 are rejected under 35 U.S.C. 103(a) as being unpatentable over Day (US 6243722) and Merritt (US 6041335);

a) claim 1 - Examiner states Day teaches:

- a comment review system of storing and managing a set of comments associated with a source file (Day abstract, also column 9 lines 13-24; compare with claim 1 "A file review system ... with a source file, comprising ..."

response: Day does not teach management of comments in the same manner as does Applicant. Applicant manages comments so they may be inserted into the document in context for display.

- accepting data and displaying a source file as an HTML file (Day Figure 6 items 130, 134; compare with claim 1 "means for accepting data from the source file ... source file as a markup file,")

response: Day is referring to taking either an HTML file or a flat text file and storing it in an HTML format, whereas the applicant's invention deals with storing an HTML formatted document in a binary markup file, a machine-readable file used to reconstruct the HTML file at display time. This binary markup stores a linked-list representation of the HTML tags and structures allowing for quick identification of text versus non-text portions, location of comment insertion points and displayed comments, and fast text search without having to scan the non-text portions such as tags. The markup file used in the invention is of a different structure and for a different purpose not taught by Day.

- creating a comment database of comments associated with said source file (Day column 9 lines 12-24; compare with claim 1 "means for creating a comment file containing data ...comments associated with the source file")

response: Day teaches maintaining comments but does not teach maintaining comments so that they may be retrieved for contextual display. Applicant teaches comment and source file association via line numbering to ensure correct placement in generated output.

- input and acceptance of new comments via pop-up window into a comment database (Day Abstract, also Figure 8 and column 9 lines 12-24; compare with claim 1 "means for accepting new comments for inclusion ... to correspond to the complete set of comments")

response: Applicant's invention does not use the pop-up window for comment input and acceptance. Applicant's invention teaches comment input and acceptance within a view of the document itself in the position where the comment will appear in the document once entered (in proper context), so users can see the context in which they are entering their comments. Day does not teach this aspect.

- Day does not specifically teach generation of hypertext document said document including portions corresponding to associated comments. However Merritt teaches hypertext document passed to various users, said document also incorporating comments (Merritt Figure 3, also column 5 lines 1-6, 64-67, column 6 lines 1-17; compare with claim 1 "means for generating a hypertext document from the markup file ... the set of comments associated with the source file"). It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Merritt to Day, because Merritt's taught advantage of including comments within a document, providing users of Day a away to inspect and comment upon previous comments made to a document.

response: Both Merritt and Day teach including graphic or textual indicators of the existence of comments within documents; they do not teach including comments themselves in documents. In both Merritt and Day the user must effect an action on a glyph or other indicator, from the reviewable content displayed, to view particular comments. Thus the objection that

Merritt taught inclusion of comments in the document is not valid, as Merritt only taught inclusion of an indication of the existence of comments in the document.
- display of a reviewable hypertext document (Day Figure 6; compare with claim 1 "means for communicating the hypertext document to a user for display.")

response: Applicant displays comments within their proper document context without user intervention, Day does not. Amendment proposed as shown.

b) claim 9 - Examiner states Day teaches CGI (Day column 5 lines 49-54, 58-62; compare with claim 9)

c) claim 10 - Examiner states Day teaches a solicited review button requiring user input at designated review points in a document, said comments stored in a comment database (Day column 7 lines 60-64, column 9 lines 12-24; compare with claim 10)

response: In Day comment insertion markers are placed at points designated manually or by template, by the author and is not really related to the instant claim. This claim particularly relates to large documents being automatically broken up by the review system into smaller, logically grouped sections and the review system displaying the comments that apply to sentences and lines within that section in line with the review document text. In the applicant's invention comment insertion markers are automatically placed wherever a sentence is determined to end, in free-flowing text, or whenever a line ends, in text that does not wrap at margins. This invention does not require user action based on a review button to ascertain where comments are to be placed. (see application page 21 lines 14 to page 23 line 13)

d) claim 12 - Examiner states Day teaches a comment input box comprising various levels of comment importance "severity" (Day Figure 8 item 162, items Comment, Typo, Problem, and Issue; compare with claim 12)

e) claim 16 - Examiner states Day does not specifically teach restricting to a specific set

of users. However, this limitation would have been obvious to one of ordinary skill in the art at the time of the invention, in view of Day, because Day teaches a Receiver and Register button for a comment (Day Figure 8 under item 162; compare with claim 16)

response: The Register button in Day Figure 8 is not supported in the description provided by Day therefore Examiner cannot suggest its intent as such. Applicant believes the Register button may allow users to enter their names so that comments may be associated with them. This technique affords simple identification and does not constitute selective restriction. Applicant's claim relates specifically to the document author's ability to choose who is able to enter comments. Some users may have the authority to review the document without being able to enter comments. (see application page 37 lines 9 to page 38 line 6)

f) claim 17 - Examiner states Day does not specifically teach restricting comments to a due date. However, Merritt teaches blinking notes associated with timer messages (Merritt Figure 13; compare with claim 17). It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Merritt to Day, because of Merritt's taught advantage of timed messages, providing the user of Day a length of time associated with a comment.

response: This is not what Merritt teaches. Merritt teaches the use of blinking of glyphs so that they are easily identified on a complex graphical image. The timed message described in Merritt actually relates to Windows Application Programmer Interface messages to the Windows operating system, instructing it to turn a glyph on and off at selected intervals in order to make it blink. Merritt's claims and descriptions relating to Merritt Figure 13 have no bearing on the feature of disabling a comment entry after an author chosen cutoff date has passed.

g) claim 18,19,20 - Examiner states Day does not specifically teach user sequence order, or statistics. However, Merritt teaches sending a document to a list of users sequentially, utilizing a statistical routing list (Merritt Figure 6, column 5 lines 64-67, column 6 lines 1-17; compare with claim 18, 19, 20). It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Merritt to Day, because of Merritt's taught advantage of sequential

commenting, providing the user of Day the capability of commenting upon other users comments, and of using routing statistics to map the path of a document.

response: Applicant's **claim 18** teaches navigating the comment file in user-defined sequence order. This navigation feature provides users the ability to specify via comment filters which comments will be shown in a specific document view and only documents containing comments matching those filters are to be shown. (see application page 33 lines 1-5 and page 36 line 18 to page 37 line 6).

Applicant's **claim 19** teaches the display of statistics related to a set of comments and has no bearing on the use in Merritt of the routing list reference cited wherein Merritt teaches which reviewer next access a document. Applicant's claim 19 (for more information see application page 40 lines 1-12) relates to displaying statistics (comments by disposition and type; for each user, comments created or appended to) for comments in the review. The cited reference provided by the Examiner to Merritt Figure 6, column 5 lines 64-67, column 6 lines 1-17 do not teach use of routing statistics, rather the material describes a simple routing list. The Examiner further cites Merritt's taught advantage of sequential commenting, wherein the Applicant's invention deals with simultaneous commenting. While the distinction between sequential and simultaneous commenting is not the essence of claim 19, it does point another difference between the instant invention and the cited prior art.

Applicant's **claim 20** teaches the use of hypertext links in the Statistics page to lead to a list of all files containing comments whose attributes match those of the statistics represented. (see application page 40 lines 5-12). This teaching is not described by either Day or Merritt.

h) claim 33 - Examiner states Day teaches:

- a comment review system of storing and managing a set of comments associated with a source file (Day abstract, also column 9 lines 13-24; compare with claim 33 "A method of storing and managing a set of comments ... the method comprising the steps of ..."

response: Day does not teach management of comments in the same manner as does Applicant. Applicant manages comments so they may be inserted into the document in context

for display.

- accepting data and displaying a source file as an HTML file (Day Figure 6 items 130, 134; compare with claim 33 "accepting data from the source file ... source file as a markup file,")

response: Day is referring to taking either an HTML file or a flat text file and storing it in an HTML format, whereas the applicant's invention deals with storing an HTML formatted document in a binary markup file, a machine-readable file used to reconstruct the HTML file at display time. This binary markup stores a linked-list representation of the HTML tags and structures allowing for quick identification of text versus non-text portions, location of comment insertion points and displayed comments, and fast text search without having to scan the non-text portions such as tags. The markup file used in the invention is of a different structure and for a different purpose not taught by Day.

- creating a comment database of comments associated with said source file (Day column 9 lines 12-24; compare with claim 33 "creating a comment file containing data ...comments associated with the source file")

response: Day teaches maintaining comments but does not teach maintaining comments so that they may be retrieved for contextual display. Applicant teaches comment and source file association via line numbering to ensure correct placement in generated output.

- input and acceptance of new comments via pop-up window into a comment database (Day Abstract, also Figure 8 and column 9 lines 12-24; compare with claim 33 "responding to user input to accept new comments for inclusion ... to correspond to the complete set of comments")

response: Applicant's invention does not use the pop-up window for comment input and acceptance. Applicant's invention teaches a method where comment input and acceptance is within a view of the document itself in the position where the comment will appear in the document once entered, so that users can see the context in which they are entering their comments. Day does not teach this aspect.

- Day does not specifically teach generation of hypertext document said document including

portions corresponding to associated comments. However Merritt teaches hypertext document passed to various users, said document also incorporating comments (Merritt Figure 3, also column 5 lines 1-6, 64-67, column 6 lines 1-17; compare with claim 33 "responding to user input to dynamically generate a hypertext document from the markup file ... the set of comments associated with the source file"). It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Merritt to Day, because of Merritt's taught advantage of including comments within a document, providing users of Day a away to inspect and comment upon previous comments made to a document.

response: Both Merritt and Day teach including graphic or textual indicators of the existence of comments within documents; they do not teach including comments themselves in documents. In both Merritt and Day the user must effect an action on a glyph or other indicator, from the reviewable content displayed, to view particular comments. Thus the objection that Merritt taught inclusion of comments in the document is not valid, as Merritt only taught inclusion of an indication of the existence of comments in the document.

- display of a reviewable hypertext document (Day Figure 6; compare with claim 33 "communicating the hypertext document to a user for display.")

response: Applicant displays comments within their proper document context without user intervention, Day does not. Amendment proposed for clarity.

i) claim 40 - Examiner rejects dependent claim 40, using the same rationale as claim 33

response: Amended claim 33 is believed to overcome Examiner's objections, therefore claim 40 remains as before.

Claims 2-8, 11, 13-15, 21-32, 34-38, 39, 41 are rejected under 35 U.S.C. 103(a) as being unpatentable over Day (US 6243722) in view of Merritt (US 6041335) and further in view of Tran (US 6054990);

a) claim 2 - Examiner states Day does not specifically teach representation of a source

file as a linked list. However Tran teaches insertion of annotation text into linked list objects (Tran column 15 lines 30-39; compare with claim 2). It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Tran to Day, because of Tran's taught advantage of linked lists, providing a user of Day with a way to dynamically store files.

response: Applicant's invention teaches storing a source file (typically HTML source) as a markup file that consists of a linked list of objects corresponding to the HTML components, text and white space components, comment insertion markers, and comment display objects. Applicant asserts that Tran teaches inserting the annotation itself into the linked list. Tran's method relate to graphic objects and not to document review systems. Day does not teach storing flat text or HTML files as markup files in a similar fashion to that described by Applicant's invention. Rather, Day is referring to taking either an HTML file or a flat text file and storing it in an HTML format which is for a different purpose than the markup file stored in Applicant's invention.

b) claims 3,4,5 - Examiner states Day teaches a hypertext review button symbol, reflective of a comment insertion point (Day Figure 6 item 140, also column 7 lines 25-28; compare with claims 3,4,5)

response: claim 3 - While Day does teach displaying a hypertext review symbol, Day does not teach storing comment insertion markers in the markup file. Applicant teaches storing comment insertion objects in the linked list, generating a hypertext document for display that includes symbols indicating where comments may be inserted.

claim 4 - teaches making comment insertion symbols into hypertext links that then allow comment entry; Day does teach this

claim 5 - teaches storing the reviewable document as a linked list of objects in which comment display objects are inserted to indicate where a comment, if present, should be displayed. A comment display object is an 'invisible' object that is used, at display time, to determine where any applicable comments for a particular line number are to be displayed, and what special tags need to be turned off before, and back on after, the comment itself is displayed.

Day (in 7:25-28) teaches modifying the document review system's HTML representation of a source file to include the actual comment insertion symbols. This is not the purpose of a comment display object of the applicant's invention.

c) claim 6 - Examiner states Day teaches opening and closing of a pop-up window for comment insertion, said window opened via document review button (Day Figure 8; compare with claim 6)

response: Applicant teaches closing HTML tags before a comment is displayed, and reopening those HTML tags after a comment is displayed, so that the comment itself does not inherit the HTML structure of the surrounding text, but does not compromise that structure either. This is elaborated in application on page 22 lines 14-18 and page 33 lines 18 to page 34 line 5 and is best illustrated by example; consider an HTML source file that contains an unordered list surrounded by italic and bold start and end tags:

```
<i><b><ul>
  <li>Item 1</li>
  <li>Item 2</li>
  <li>Item 3</li>
</ul></b></i>
```

This would display in a browser as:

* ***Item 1***
* ***Item 2***
* ***Item 3***

Applicant's invention stores a comment insertion marker after each <li> object, and a comment display marker after each object. Each comment display marker includes the end and start tags that are required to display the text of the comment in such a way that it appears *outside* any structure for the item it annotates. It is the object of claim 6 and its use of closing and opening tags within comment display markers, to display:

(Preferred version)

* ***Item 1***

Comment for item 1

* *Item 2*

* *Item 3*

Rather than

(Undesirable version)

* *Item 1*

  Comment for item 1

* *Item 2*

* *Item 3*

Note that in the undesirable version, the comment appears in bold and italics, and indented within the list, while in the preferred version the comment is displayed without the special formatting of the surrounding text. The purpose as claimed in claim 6 is to track what HTML tags are currently open and must be closed prior to displaying any comment at a given location, and correspondingly to track what HTML tags must be reopened after comment display. In this context, the comment display object for the comment to be displayed after Item 1 would include these attributes:

CloseBefore (tags to close before comment display): (end the list item and the list, and turn off bold and italics)

OpenAfter (tags to open after comment display): (turn italics and bold back on, and restart the list and the list item).

Examiner's objection cites Day's teaching of "opening and closing of a pop-up window for comment insertion" but it has nothing to do with storing opening and closing tags. Day's teaching reflects a user interface action (a window opens, the user enters a comment, then the window closes). Applicant teaches storing information about the current context where a comment is to be displayed in-body, so that at display time, that context and the comments displayed there do not cause formatting problems.

d) claims 7,8 - Examiner states Day does not specifically teach representation of a source file as a linked list. However Tran teaches insertion of annotation text into linked list objects (Tran column 15 lines 30-39; compare with claims 7,8). It would have been obvious to one of

ordinary skill in the art at the time of the invention to apply Tran to Day, because of Tran's taught advantage of linked lists, providing a user of Day with a way to dynamically store and manipulate objects.

response: Claim 7 teaches *normalizing* the HTML that is stored in the linked list, so that all matching end tags are in the correct location for the start tags found in the document (e.g. every has a matching), and so that all extraneous HTML tags are removed (see application pages 18 line 18 to page 21 line 5). The advantage of using the linked list to normalize is specifically taught at application page 20 line 28 where it is stated that "The use of a linked list permits the efficient insertion and deletion of objects during the normalization step". Day does not perform any HTML normalization or describe efficient processing of HTML files, further Day does not store the files in any linked-list format but in HTML itself, whereas claim 7 specifically relates to *normalizing HTML* through a linked list, rather than adding annotation objects to a linked list as described by Tran. Therefore Applicant respectfully submits that the invention, as claimed, is not described, or taught or suggested by Day or Tran or combination of the two.

Claim 8 describes the types of objects stored in the linked list. This is not claiming that a linked list of diverse objects is unique, merely that a linked list of objects as embodied in a document review system can contain objects of these types. Applicant's invention teaches storing a source file (typically HTML source) as a markup file that consists of a linked list of objects corresponding to the HTML components, text and white space components, comment insertion markers, and comment display objects. Applicant asserts that Tran teaches inserting the annotation itself into the linked list. Tran's method relate to graphic objects and not to document review systems. Day does not teach storing flat text or HTML files as markup files in a similar fashion to that described by Applicant's invention. Rather, Day is referring to taking either an HTML file or a flat text file and storing it in an HTML format which is for a different purpose than the markup file stored in Applicant's invention. Therefore Applicant respectfully submits that the invention, as claimed, is not described, or taught or suggested by combining Tran with

Day.

e) claim 11 - Examiner states Day teaches a contents line identifier in the form of a review button on a line of document text (Day Figure 6 item 140; compare with claim 11).

response: Day teaches, in Figure 6 item 140, a solicited review comment (see Day column 7 lines 58-60) button and more correctly teaches the display of graphic elements to prompt the user to enter comments, whereas, this is not what claim 11 teaches. Applicant's claim 11 teaches the processes by which objects in the linked list are of different types and represent different parts of the HTML source file (claim 11 lines 2-3); line number identifiers are associated with comment insertion and comment display objects (claim 11 lines 4-6); comment files include line number information for each entered comment so that those comments can be displayed in the correct location when the reviewable file is displayed (claim 11 lines 7-9) and reviewable documents are displayed, including text, HTML tags, comment insertion markers, and applicable comments (claim 11 lines 10-21)

f) claims 13,14,15 - Examiner states Day teaches a comment review button on a source document, the activation of which results in the activation of a comment input window for input of a comment associated with a specific topic (Day Figures 6-8, column 7 lines 58-67; compare with claims 13,14,15).

response: Claim 13 teaches means for "selectively excluding portions of the comment file ... whereby a selectively defined subset of the set of comments is represented in the hypertext document". This relates to filtering (see application page 33 lines 1-5 and page 36 line 18 to page 37 line 6). Examiner's objection cites the "comment review button, the activation of which results in the activation of a comment input window...". which does not, in Day, teach selective display of comments based on a user's filtering criteria.

Claim 14 elaborates claim 13 by stating that comments have particular attributes that can form the basis for filtering. This element is not taught by Day in the referenced section (column 7 lines 58-67). Day does describe comment attributes such as type and severity but he does not teach the filtering of comments to determine which comments to display.

Claim 15 elaborates claim 14 by describing the comment attributes in detail. Claim 15 claims that comment filtering can be performed on a number of attributes, including those cited. This element is not taught by Day, who did not deal with filtering, and who does not provide for in-body display of comments.

g) claim 21 - Examiner states Day teaches:

- a comment review system of storing and managing a set of comments associated with a source file (Day abstract, also column 9 lines 13-24; compare with claim 21 "A web-based file review system ... with one or more webs of source file, comprising ..."

response: Day does not teach management of comments in the same manner as does Applicant. Applicant manages comments so they may be inserted into the document in context for display.

- accepting data and displaying a source file as an HTML file, the analyzation of which involves parsing (Day Figure 6 items 130, 134; compare with claim 21 "a parser to parse ... associated with one or more comments.")

response: Day does not use a linked list but uses parsing to determine where to insert specific HTML tags into the source HTML document. Applicant is claiming the parsing specifically required to create a linked list of a particular format suitable to review documents..

- Day does not specifically teach representation of a source file as a linked list. However Tran teaches insertion of annotation text into linked list objects (Tran column 15 lines 30-39; compare with claim 21 "a linked list of objects"). It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Tran to Day, because of Tran's taught advantage of linked lists, providing a user of Day with a way to dynamically store files.

response: Applicant's invention teaches storing a source file (typically HTML source) as a markup file that consists of a linked list of objects corresponding to the HTML components, text and white space components, comment insertion markers, and comment display objects. Applicant asserts that Tran teaches inserting the annotation itself into the linked list. Tran's method relate to graphic objects and not to document review systems. Day does not teach storing

flat text or HTML files as markup files in a similar fashion to that described by Applicant's invention. Rather, Day is referring to taking either an HTML file or a flat text file and storing it in an HTML format which is for a different purpose than the markup file stored in Applicant's invention. Therefore Applicant respectfully submits that the invention, as claimed, is not described, or taught or suggested by Tran, Day or a combination.

- creating a comment database of comments associated with said source file (Day column 9 lines 12-24; compare with claim 21 "a set of comment files ...updating the associated comment file")

response: Day teaches maintaining comments but does not teach maintaining comments so that they may be retrieved for contextual display. Applicant teaches comment and source file association via line numbering to ensure correct placement in generated output.

- input and acceptance of new comments via pop-up window into a comment database, said invention utilizing CGI (Day Abstract, also Figure 8 and column 9 lines 12-24, column 5 lines 50-55; compare with claim 21 "common gateway interface program ... reviewer-selected source file", and "common gateway interface program code means for generating a hypertext document ... the reviewer-selected source files")

response: Day teaches use of CGI to deliver program function. Applicant's method does not make use of pop-up windows, instead uses placement of comments in line. This claim deals with generating a representation of the reviewable document, including comments in line from the markup files and comment files. Day does not teach this aspect, nor does Day teach display of comments in line (nor does Merritt).

- Day does not specifically teach generation of hypertext document said document including portions corresponding to associated comments. However Merritt teaches hypertext document passed to various users, said document also incorporating comments associated with positioned icons (Merritt Figure 3, also column 5 lines 1-6, 64-67, column 6 lines 1-17; compare with claim 21 "the hypertext document including ... comment display objects", and "the hypertext document selectively including ... for accepting new comments"). It would have been obvious to one of

ordinary skill in the art at the time of the invention to apply Merritt to Day, because Merritt's taught advantage of including comments within a document, providing users of Day a away to inspect and comment upon previous comments made to a document.

response: Both Merritt and Day teach including graphic or textual indicators of the existence of comments within documents; they do not teach including comments themselves in documents. In both Merritt and Day the user must effect an action on a glyph or other indicator, from the reviewable content displayed, to view particular comments. Thus the objection that Merritt taught inclusion of comments in the document is not valid, as Merritt only taught inclusion of an indication of the existence of comments in the document.

- hypertext links resulting in a display of comment input form (Day Figures 6-8; compare with claim 21 "the hypertext document selectively ... reviewer entry of comments").

response:. Applicant's method does not make use of pop-up windows, instead uses placement of comments in line. This claim deals with generating a representation of the reviewable document, including comments in line from the markup files and comment files. Day does not teach this aspect, nor does Day teach display of comments in line (nor does Merritt).

- display of a reviewable hypertext document (Day Figure 6; compare with claim 21 "means for communicating the hypertext document to a user for display.")

response: Applicant displays comments within their proper document context without user intervention, Day does not. Amendment proposed as shown.

h) claims 22,23,24 - Examiner states Day teaches a hypertext review button (requiring opening and closing tags) for opening/closing a comment window associated with a hypertext document (Day Figure 8). Day also teaches a hypertext document with various hypertext review buttons at various locations in said document and buttons requiring the use of various tags for location placement, functions, and display types (Day Figures 6-8; compare with claims 22,23,24).

response: Day's teaching reflects a user interface action (a window opens, the user enters a comment, then the window closes). Applicant teaches storing information about the current

context where a comment is to be displayed in line, so that at display time, that context and the comments displayed there do not cause formatting problems. Applicant claim 22 deals with close-before and open-after tag values that are maintained for each comment display object, so that when a comment needs to be displayed at a location, the active HTML tags are closed off before the comment, and reopened afterwards (see example in response to the objections to claim 6 previously provided)

Claim 23 The applicant's use of linked-list objects have types relating to their function in the document to be displayed, namely, text, white space, tag, end-tag, and comment-related objects. The applicant's use of linked list objects bear no resemblance to Day's use of review buttons.

Claim 24 claims and describes the Applicant's method by which comments are merged into the linked list retrieved from the markup file, at the points that correlate comment line numbers with the line numbers stored in comment display markers. Day does not teach displaying comments in line, nor a method by which they might so be displayed.

i) claim 25 - Examiner states Day teaches:

- a comment review system of storing and managing a set of comments associated with a source file (Day abstract, also column 9 lines 13-24; compare with claim 25 "a computer usable medium ... in the article of manufacture, comprising ..."

response: Day does not teach management of comments in the same manner as does Applicant. Applicant manages comments so they may be inserted into the document in context for display.

- accepting data and displaying a source file as an HTML file, the analyzation of which involves parsing (Day Figure 6 items 130, 134; compare with claim 25 "to parse a selected one of the set ... associated with one or more comments.")

response: Day does not use a linked list and Day uses parsing to determine where to insert specific HTML tags into the source HTML document. Applicant is claiming the parsing specifically required to create a linked list of a particular format suitable to review documents.

- Day does not specifically teach representation of a source file as a linked list. However Tran teaches insertion of annotation text into linked list objects (Tran column 15 lines 30-39; compare with claim 25 "a linked list of objects"). It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Tran to Day, because of Tran's taught advantage of linked lists, providing a user of Day with a way to dynamically store files.

response: Applicant's invention teaches storing a source file (typically HTML source) as a markup file that consists of a linked list of objects corresponding to the HTML components, text and white space components, comment insertion markers, and comment display objects. Applicant asserts that Tran teaches inserting the annotation itself into the linked list. Tran's method relates to graphic objects and not to document review systems. Day does not teach storing flat text or HTML files as markup files in a similar fashion to that described by Applicant's invention. Rather, Day is referring to taking either an HTML file or a flat text file and storing it in an HTML format which is for a different purpose than the markup file stored in Applicant's invention. Therefore Applicant respectfully submits that the invention, as claimed, is not described, or taught or suggested by Tran, Day or a combination.

- creating a comment database of comments associated with said source file (Day column 9 lines 12-24; compare with claim 25 "a set of comment files ...update the associated comment file")

response: Day teaches maintaining comments but does not teach maintaining comments so that they may be retrieved for contextual display. Applicant teaches comment and source file association via line numbering to ensure correct placement in generated output.

- input and acceptance of new comments via pop-up window into a comment database, said invention utilizing CGI (Day Abstract, also Figure 8 and column 9 lines 12-24, column 5 lines 50-55; compare with claim 25 "generate a hypertext document from a markup file ... reviewer-selected source file")

response: Day teaches use of CGI to deliver program function. Applicant's method does

not make use of pop-up windows, instead uses placement of comments in line. This claim deals with generating a representation of the reviewable document, including comments in line from the markup files and comment files. Day does not teach this aspect, nor does Day teach display of comments in line (nor does Merritt).

- Day does not specifically teach generation of hypertext document said document including portions corresponding to associated comments. However Merritt teaches hypertext document passed to various users, said document also incorporating comments associated with positioned icons (Merritt Figure 3, also column 5 lines 1-6, 64-67, column 6 lines 1-17; compare with claim 25 "the hypertext document including ... comment display objects", and "the hypertext document selectively including ... for accepting new comments"). It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Merritt to Day, because Merritt's taught advantage of including comments within a document, providing users of Day a way to inspect and comment upon previous comments made to a document.

response: Both Merritt and Day teach including graphic or textual indicators of the existence of comments within documents; they do not teach including comments themselves in line in documents. In both Merritt and Day the user must effect an action on a glyph or other indicator, from the reviewable content displayed, to view particular comments. Thus the objection that Merritt taught inclusion of comments in the document is not valid, as Merritt only taught inclusion of an indication of the existence of comments in the document.

- hypertext links resulting in a display of comment input form (Day Figures 6-8; compare with claim 25 "the hypertext document selectively ... reviewer entry of comments").

response: Applicant teaches modification of the HTML generated from the markup file, to include appropriate CGI links so any link the user follows allows the CGI to intercept that link. This teaching is not evident in Day as cited.

- display of a reviewable hypertext document, as well as CGI (Day Figure 6; compare with claim 25 "common gateway interface" and "communicate the hypertext document to a user for

display.")

response: Applicant displays comments within their proper document context without user intervention, Day does not. Amendment proposed as shown.

j) claims 26,27,28 - Reflect the article of manufacture comprising computer executable instructions for performing the methods of the system as claimed in claims 22, 23, 24 and are rejected along the same rationale.

response: Since objections to claims 22, 23, 24, are believed to have been overcome, applicant has not rewritten dependent claims 26,27,28.

k) claim 29 - Examiner states Day teaches:

- a comment review system of storing and managing a set of comments associated with a source file (Day abstract, also column 9 lines 13-24; compare with claim 29 "a computer usable medium ... the computer program product having ..."

response: Day does not teach management of comments in the same manner as does Applicant. Applicant manages comments so they may be inserted into the document in context for display.

- accepting data and displaying a source file as an HTML file, the analyzation of which involves parsing (Day Figure 6 items 130, 134; compare with claim 29 "to parse a selected one of the set ... associated with one or more comments.")

response: Day does not teach the use a linked list and Day uses parsing to determine where to insert specific HTML tags into the source HTML document. Day does not teach this type of markup creation. Applicant is claiming the parsing specifically required to create a linked list of a particular format suitable to review documents. Teachings of Tran for linked list use do not relate to this claim as Tran inserts comments into the linked list.

- Day does not specifically teach representation of a source file as a linked list. However Tran teaches insertion of annotation text into linked list objects (Tran column 15 lines 30-39; compare with claim 29 "a linked list of objects"). It would have been obvious to one of ordinary skill in

the art at the time of the invention to apply Tran to Day, because of Tran's taught advantage of linked lists, providing a user of Day with a way to dynamically store files.

response: Applicant's invention teaches storing a source file (typically HTML source) as a markup file that consists of a linked list of objects corresponding to the HTML components, text and white space components, comment insertion markers, and comment display objects. Applicant asserts that Tran teaches inserting the annotation itself into the linked list. Tran's method relates to graphic objects and not to document review systems. Day does not teach storing flat text or HTML files as markup files in a similar fashion to that described by Applicant's invention. Rather, Day is referring to taking either an HTML file or a flat text file and storing it in an HTML format which is for a different purpose than the markup file stored in Applicant's invention. Therefore Applicant respectfully submits that the invention, as claimed, is not described, or taught or suggested by Tran, Day or a combination.

- creating a comment database of comments associated with said source file (Day column 9 lines 12-24; compare with claim 29 "a set of comment files ... update the associated comment file")

response: Day teaches maintaining comments but does not teach maintaining comments so that they may be retrieved for contextual display. Applicant teaches comment and source file association via line numbering to ensure correct placement in generated output.

- input and acceptance of new comments via pop-up window into a comment database, said invention utilizing CGI (Day Abstract, also Figure 8 and column 9 lines 12-24, column 5 lines 50-55; compare with claim 29 "generate a hypertext document from a markup file ... reviewer-selected source file")

response: Day teaches use of CGI to deliver program function. Applicant's method does not make use of pop-up windows, instead uses placement of comments in line. This claim deals with generating a representation of the reviewable document, including comments in line from the markup files and comment files. Day does not teach this aspect of generating HTML from a

markup file, nor does Day teach display of comments in line (nor does Merritt).

- Day does not specifically teach generation of hypertext document said document including portions corresponding to associated comments. However Merritt teaches hypertext document passed to various users, said document also incorporating comments associated with positioned icons (Merritt Figure 3, also column 5 lines 1-6, 64-67, column 6 lines 1-17; compare with claim 29 "the hypertext document including ... comment display objects", and "the hypertext document selectively including ... for accepting new comments"). It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Merritt to Day, because Merritt's taught advantage of including comments within a document, providing users of Day a way to inspect and comment upon previous comments made to a document.

response: Both Merritt and Day teach including graphic or textual indicators of the existence of comments within documents; they do not teach including comments themselves in line in documents. In both Merritt and Day the user must effect an action on a glyph or other indicator, from the reviewable content displayed, to view particular comments. Thus the objection that Merritt taught inclusion of comments in the document is not valid, as Merritt only taught inclusion of an indication of the existence of comments in the document.

- hypertext links resulting in a display of comment input form (Day Figures 6-8; compare with claim 29 "the hypertext document selectively ... reviewer entry of comments").

response: Applicant teaches modification of the HTML generated from the markup file, to include appropriate CGI links so any link the user follows allows the CGI to intercept that link. This teaching is not evident in Day nor Merritt as cited referring to their use of comment insertion markers (buttons).

- display of a reviewable hypertext document, as well as CGI (Day Figure 6; compare with claim 29 "common gateway interface" and "communicate the hypertext document to a user for display.")

response: Applicant displays comments within their proper document context without

user intervention, Day does not. Amendment proposed as shown.

l) claims 30,31, 32 - Examiner states these claims reflect the computer program product comprising computer executable instructions for performing the methods of the system as claimed in claims 22, 23, 24 and are rejected along the same rationale.

response: Applicant believes each of the objections and rejections regarding claims 22, 23, 24 have been addressed and respectfully submits that these claims pending herein are now allowable using similar rationale.

m) claims 34, 35, 36, 37, 38 - Examiner states these claims incorporate substantially similar subject matter as claimed in claims 2, 3, 4, 5, 6, respectively and are rejected along the same rationale.

response: Applicant believes each of the objections and rejections regarding claims 2, 3, 4, 5, 6 have been addressed and respectfully submits that these claims pending herein are now allowable using similar rationale.

n) claim 39 - Examiner states Day teaches:

- a comment review system of storing and managing a set of comments associated with a source file (Day abstract, also column 9 lines 13-24; compare with claim 39 "A method of storing ... comprising the steps of "

response: Day does not teach management of comments in the same manner as does Applicant. Applicant manages comments so they may be inserted into the document in context for display.

- accepting data and displaying a source file as an HTML file, the analyzation of which involves parsing (Day Figure 6 items 130, 134; compare with claim 39 "parsing a selected one of the set ... associated with one or more comments.")

response: Day does not teach the use a linked list and Day uses parsing to determine where to insert specific HTML tags into the source HTML document. Day does not teach this type of markup creation. Applicant is claiming the parsing specifically required to create a linked

list of a particular format suitable to review documents. Teachings of Tran for linked list use do not relate to this claim as Tran inserts comments into the linked list itself.

- Day does not specifically teach representation of a source file as a linked list. However Tran teaches insertion of annotation text into linked list objects (Tran column 15 lines 30-39; compare with claim 39 "a linked list of objects"). It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Tran to Day, because of Tran's taught advantage of linked lists, providing a user of Day with a way to dynamically store files.

response: Applicant's invention teaches storing a source file (typically HTML source) as a markup file that consists of a linked list of objects corresponding to the HTML components, text and white space components, comment insertion markers, and comment display objects. Applicant asserts that Tran teaches inserting the annotation itself into the linked list. Tran's method relates to graphic objects and not to document review systems. Day does not teach storing flat text or HTML files as markup files in a similar fashion to that described by Applicant's invention. Rather, Day is referring to taking either an HTML file or a flat text file and storing it in an HTML format which is for a different purpose than the markup file stored in Applicant's invention. Therefore Applicant respectfully submits that the invention, as claimed, is not described, or taught or suggested by Tran, Day or a combination.

- creating a comment database of comments associated with said source file (Day column 9 lines 12-24; compare with claim 39 "a set of comment files ... update the associated comment file")

response: Day teaches maintaining comments but does not teach maintaining comments so that they may be retrieved for contextual display. Applicant teaches comment and source file association via line numbering to ensure correct placement in generated output.

- input and acceptance of new comments via pop-up window into a comment database, said invention utilizing CGI (Day Abstract, also Figure 8 and column 9 lines 12-24, column 5 lines 50-55; compare with claim 39 "on review request ... one of the set of source files" and "dynamically generating a hypertext document from a markup file ... reviewer-selected source

file").

response: While Day teaches use of CGI to deliver program function, Applicant's method does not make use of pop-up windows, instead uses placement of comments in line. This claim deals with generating a representation of the reviewable document, causing an input form to appear at the exact location in the review document where the comment will appear once entered Day does not teach this aspect of generating HTML from a markup file, nor does Day teach display of comments in line (nor does Merritt). Applicant teaches an improvement over both Day and Merritt.

- Day does not specifically teach generation of hypertext document said document including portions corresponding to associated comments. However Merritt teaches hypertext document passed to various users, said document also incorporating comments associated with positioned icons (Merritt Figure 3, also column 5 lines 1-6, 64-67, column 6 lines 1-17; compare with claim 39 "the hypertext document including ... comment display objects", and "the hypertext document selectively including ... for accepting new comments"). It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Merritt to Day, because Merritt's taught advantage of including comments within a document, providing users of Day a way to inspect and comment upon previous comments made to a document.

response: Both Merritt and Day teach including graphic or textual indicators of the existence of comments within documents; they do not teach including comments themselves in line in documents. In both Merritt and Day the user must effect an action on a glyph or other indicator, from the reviewable content displayed, to view particular comments. Thus the objection that Merritt taught inclusion of comments in the document is not valid, as Merritt only taught inclusion of an indication of the existence of comments in the document.

- hypertext links resulting in a display of comment input form (Day Figures 6-8; compare with claim 39 "the hypertext document selectively ... reviewer entry of comments" and "hypertext links").

response: Applicant teaches modification of the HTML generated from the markup file, to include appropriate CGI links so any link the user follows allows the CGI to intercept that link. This teaching is not evident in Day nor Merritt as cited referring to their use of comment insertion markers (buttons).

- display of a reviewable hypertext document, as well as CGI (Day Figure 6 column 5 lines 50-54; compare with claim 39 "common gateway interface" and "communicate the hypertext document to a user for display.")

response: Applicant displays comments within their proper document context without user intervention, Day does not. Amendment proposed as shown.

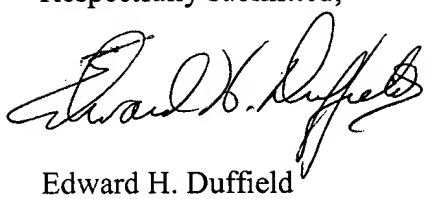
o) claim 41 - Examiner states claim 41 reflects the computer program product comprising computer executable instructions for performing the methods as claimed in claim 39 and is rejected along the same rationale.

response: Applicant believes each of the objections and rejections regarding claim 39 has been addressed and respectfully submits that this claim pending herein is now allowable using similar rationale.

Applicant has reviewed prior art made of record and not relied on, and does not feel it is pertinent to this instant application.

In view of the foregoing amendments and discussion, it is respectfully submitted that the claims as now remaining for examination are allowable over the art and rejections posed , wherefore the Examiner's reconsideration of the claims as amended and in light of the remarks is respectfully requested.

Respectfully submitted,



Edward H. Duffield
Attorney for Applicants
Reg. No. 25,970

EHD:jdl

Phone: 919-254-1301
Fax: 919-254-4330

Version with Markings to Show Changes Made

In the Title:

The title has been changed as follows:

Web-Based File Review System for Displaying Comments Inlaid in a Document.

In the Abstract:

The abstract has been changed as follows:

A system for reviewing files which permits comments to be inserted in files to be viewed with a hypertext browser. When the hypertext mark-up language employed is HTML, text files are converted to an HTML representation. An HTML file is represented by a linked list of objects. Comment insertion markers and comment display objects are inserted at predefined points in the HTML linked list representation. The linked list is stored as a binary file and has a comment file associated with it. Access to the HTML file by reviewers and authors causes the regeneration of the HTML document by a Common Gateway Interface which recreates the linked list representation of the document from the binary file and which then generates HTML code from the linked list. Comments may be entered by reviewers working in parallel on the HTML document. Comments are displayed inlaid at the next regeneration of the HTML document by the system.

In the Claims:

Claims 9 and 12 have been canceled without prejudice.

Claim 1 has been amended as follows:

1. (amended) A file review system for storing and managing a set of comments associated

with a source file, comprising

means for accepting data from the source file and storing a representation of the source file as a markup file,

means for creating a comment file containing data representing the set of comments associated with the source file,

means for accepting new comments for inclusion in the set of comments associated with the source file and for updating the comment file to correspond to the complete set of comments,

means for generating a hypertext document from the markup file and from the comment file, the hypertext document corresponding to the source file and including portions corresponding to one or more of the set of comments associated with the source file,

means for communicating the hypertext document to a [user for display] browser for displaying said inlaid comments.

Claim 21 has been amended as follows:

21. (amended) A web-based file review system for storing and managing comments from a plurality of reviewers, the comments being associated with one or more webs of source files, comprising

a parser to parse a selected one of the set of source files into a linked list of objects corresponding to a hypertext representation of the selected source file, the linked list further comprising comment insertion objects and comment display objects, the parser writing the linked list of objects to a binary markup file representing the linked list of objects and corresponding to the selected one of the set of source files, each comment display object being capable of being associated with one or more comments,

a set of comment files, each comment file being associated with a one of the set of source files and comprising data representing comments associated with the one of the set of

source files,

common gateway interface program code means for accepting new comments for inclusion in the set of comments associated with a reviewer-defined source file and for updating the associated comment file,

common gateway interface program code means for generating a hypertext document from a markup file corresponding to reviewer-selected source file and from the associated comment file, the hypertext document corresponding to the reviewer-selected source file and

the hypertext document including portions corresponding to one or more of the set of comments associated with the reviewer-selected source file, the hypertext data for each portion relating to a comment to be displayed being defined by the associated comment display object,

the hypertext document selectively including hypertext links representing comment insertion objects, the hypertext links providing reviewers with forms for reviewer entry of comments,

the hypertext document selectively including hypertext data for calling the common gateway interface program for generating a hypertext document and the hypertext document selectively including hypertext data for calling the common gateway interface program for accepting new comments,

means for communicating the hypertext document to a browser for [display] <u>displaying said inlaid comments</u>.

Claim 25 has been amended as follows:

25. (amended) An article of manufacture comprising:

a computer usable medium having computer readable program code means embodied therein for causing the storage and management of comments in a web-based file review system, the comments being from a plurality of reviewers, and being associated with one or more webs of source files, the computer readable program code means in the article of manufacture comprising

computer readable program code means for causing a computer to parse a selected one of the set of source files into a linked list of objects corresponding to a hypertext representation of the selected source file, the linked list further comprising comment insertion objects and comment display objects, the parser writing the linked list of objects to a binary markup file representing the linked list of objects and corresponding to the selected one of the set of source files, each comment display object being capable of being associated with one or more comments,

computer readable program code means for causing the computer to create and manage a set of comment files, each comment file being associated with a one of the set of source files and comprising data representing comments associated with the one of the set of source files,

computer readable program code means for causing the computer to accept new comments for inclusion in the set of comments associated with a reviewer-defined source file and to update the associated comment file,

computer readable program code means for causing the computer to generate a hypertext document from a markup file corresponding to reviewer-selected source file and from the associated comment file, the hypertext document corresponding to the reviewer-selected

source file and

the hypertext document including portions corresponding to one or more of the set of comments associated with the reviewer-selected source file, the hypertext data for each portion relating to a comment to be displayed being defined by the associated comment display object,

the hypertext document selectively including hypertext links representing comment insertion objects, the hypertext links providing reviewers with forms for reviewer entry of comments,

the hypertext document selectively including hypertext data for calling the common gateway interface program for generating a hypertext document and the hypertext document selectively including hypertext data for calling the common gateway interface program for accepting new comments,

computer readable program code means for [causing the computer to communicate the hypertext document to a browser for display] communicating the hypertext document to a browser for displaying said inlaid comments.

Claim 29 has been amended as follows:

29. (amended) A computer program product for use with a hypertext server, the computer program product comprising:
a computer usable medium having computer readable program code means embodied in the medium for causing the storage and management of comments in a web-based file review system, the comments being from a plurality of reviewers, and being associated with one or more webs of source files, the computer program product having:

computer readable program code means for causing a computer to parse a selected one of the set of source files into a linked list of objects corresponding to a hypertext representation of the selected source file, the linked list further comprising comment insertion objects and comment display objects, the parser writing the linked list of objects to a binary markup file representing the linked list of objects and corresponding to the selected one of the set of source files, each comment display object being capable of being associated with one or more comments,

computer readable program code means for causing the computer to create and manage a set of comment files, each comment file being associated with a one of the set of source files and comprising data representing comments associated with the one of the set of source files,

computer readable program code means for causing the computer to accept new comments for inclusion in the set of comments associated with a reviewer-defined source file and to update the associated comment file,

computer readable program code means for causing the computer to generate a hypertext document from a markup file corresponding to reviewer-selected source file and from the associated comment file, the hypertext document corresponding to the reviewer-selected source file and

the hypertext document including portions corresponding to one or more of the set of comments associated with the reviewer-selected source file, the hypertext data for each portion relating to a comment to be displayed being defined by the associated comment display object,

the hypertext document selectively including hypertext links representing comment

insertion objects, the hypertext links providing reviewers with forms for reviewer entry of comments,

the hypertext document selectively including hypertext data for calling the common gateway interface program for generating a hypertext document and the hypertext document selectively including hypertext data for calling the common gateway interface program for accepting new comments,

computer readable program code means for causing the computer to communicate the hypertext document to a browser for [display] displaying said inlaid comments.

Claim 33 has been amended as follows:

33. (amended) A method of storing and managing a set of comments associated with a source file, in a file review system, the method comprising the steps of

accepting data from the source file and storing a representation of the source file as a markup file,

creating a comment file containing data representing the set of comments associated with the source file,

responding to user input to accept new comments for inclusion in the set of comments associated with the source file and updating the comment file to correspond to the complete set of comments,

responding to user input to dynamically generate a hypertext document from the markup file and from the comment file, the hypertext document corresponding to the source file

and including portions corresponding to one or more of the set of comments associated with the source file,

communicating the hypertext document to a [user for display] browser for displaying said inlaid comments.

Claim 39 has been amended as follows:

39. (amended) A method for storing and managing comments in a web-based file review system, the comments being from a plurality of reviewers and being associated with one or more webs of source files, comprising the steps of

parsing a selected one of the set of source files into a linked list of objects corresponding to a hypertext representation of the selected source file, the linked list further comprising comment insertion objects and comment display objects, the parser writing the linked list of objects to a binary markup file representing the linked list of objects and corresponding to the selected one of the set of source files, each comment display object being capable of being associated with one or more comments,

on review request, accepting new comments for inclusion in the set of comments associated with a reviewer-defined source file and for updating an associated comment file, the comment file being associated with a one of the set of source files and comprising data representing comments associated with the one of the set of source file,

dynamically generating a hypertext document from a markup file corresponding to reviewer-selected source file and from the associated comment file, the hypertext document corresponding to the reviewer-selected source file and

the hypertext document including portions corresponding to one or more of the set of comments associated with the reviewer-selected source file, the hypertext data for each portion relating to a comment to be displayed being defined by the associated comment display object,

the hypertext document selectively including hypertext links representing comment insertion objects, the hypertext links providing reviewers with forms for reviewer entry of comments,

the hypertext document selectively including hypertext data for calling the common gateway interface program for generating a hypertext document and the hypertext document selectively including hypertext data for calling the common gateway interface program for accepting new comments,

communicating the hypertext document to a browser for [display] displaying said inlaod comments.